DOCUMENT RESUME

ED 087 460 IR 000 204

AUTHOR	Hansen, R. W.; Shostak, A. D.
TITLE	How To Evaluate and Select Software.
PUB DATE	Apr 73
NOTE	4p.; Paper presented at the Association for Educational Data Systems Annual Convention (New Orleans, Louisiana, April 16 through 19, 1973)
EDRS PRICE	MF-$0.65 HC-$3.29
DESCRIPTORS	*Computer Programs; *Cost Effectiveness; *Evaluation; Guidelines; Guides; *Program Evaluation; *Selection
IDENTIFIERS	AEDS; Association for Educational Data Systems

ABSTRACT

Several factors relevant to the evaluation and selection of cost-effective computer software are discussed. Topics considered include: usage rights, disclosure privileges, delivery and warranty terms, maintenance agreements, program releases and modifications, installation, and remote versus on-site usage. (PB)

ED 087160

U.S. DEPARTMENT OF HEALTH, EDUCATION & WELFARE
NATIONAL INSTITUTE OF EDUCATION
THIS DOCUMENT HAS BEEN REPRODUCED EXACTLY AS RECEIVED FROM THE PERSON OR ORGANIZATION ORIGINATING IT. POINTS OF VIEW OR OPINIONS STATED DO NOT NECESSARILY REPRESENT OFFICIAL NATIONAL INSTITUTE OF EDUCATION POSITION OR POLICY

HOW TO EVALUATE AND SELECT SOFTWARE

R. W. Hansen

and

A. D. Shostak

Control Data Corporation

INTRODUCTION

The session abstract given in the program indicates that guidelines will be presented to simplify the task of evaluation and selection of software. Do not expect too much in the form of guidelines.

The single most significant issue may well be that the Computing Center is being held more and more accountable for its performance. "The price/performance ratio" of computing centers is today a very real measure of efficiency. Thus, all avenues of cost effectiveness are being explored by Center Management. This quest for cost-consciousness is taking many forms, one of which is the use of program packages obtained from a vendor for a price.

There is then at present a definite and discernible attitude that is diametrically different from the old "not-invented-here" syndrome often encountered in data-processing shops of the Sixties. It used to be that no one seriously considered any software that wasn't generated in-house or supplied "free" by the computer manufacturer. Often, what was supplied free wasn't all it was supposed to be. But users are recognizing that there is a cost to software, and "freebees" just aren't a part of anyone's game plan any longer. So, faced with a decision to "make or buy", the Center Manager is beginning to look around and see what's best.

I would like to begin by commenting on just a few of the legal considerations. Programs, as you may or may not know, are not in general patentable or copyright-able. With regard to patentability, the law has been reluctant to recognize rights in abstraction and ideas; thus, more ideas are not patentable and must be reduced to practice before rights are recognized. (1, p. 105) The Statutory Copyright Law protects the "writings of an author" from unauthorized copying. It applies to the form of expression and not to ideas that may be presented in the work. To

---

RANDALL W. HANSEN is Education Industry Manager for Control Data Corporation. He was previously employed by General Motors' Research Staff and Burroughs Corporation. He holds a Bachelor of Science Degree from the University of Wisconsin.

ANTHONY D. SHOSTAK holds an undergraduate degree in Mathematics and a graduate degree in Management Science. His twelve years in the Computer Industry include five in the Education Industry. His current position and responsibilities include those of Manager, Computer Based Education and training Marketing Support.

HR 000 904

copyright a work, it is only necessary to place the requisite copyright mark upon the "writing" and to register the material with the Copyright Office shortly after first publication. The protection is not in effect until publication takes place.(1, p. 108)

How then are the rights of the originator of a program protected? The present method of protection is based on the "trade secret" law. Where theft of a trade secret may be proved, the court can take appropriate action. A trade secret is defined very broadly as "any formula, pattern, device or compilation of information which is used in one's business, and which gives him an opportunity to obtain an advantage over competitors who do not know or use it". (1, p. 106)

The accepted method of implementing the practice of trade secret protection as it applies to computer programs is through a License-to-Use. Such a license usually has two forms: 1) the use is for a specified period of time, or 2) the use is forever. The financial arrangement is often a monthly fee in the first case and a "paid-up" or full-payment lease in the second case.

Often, under a license agreement, title to the program does not transfer, no copies may be made, the seller retains resale rights and the right to retain the program. Further, a non-disclosure clause is often a part of the license. This simply means it must not be disclosed or given away to others. It includes documentation, processing algorithm or anything else proprietary to the program. The important point is that acquistion of a software package is a legal as well as business issue.

The most important section of the contract is the bill of particulars that explicitly describes what it is you get. Included are items such as type and quantity of documentation, training, and conversion/installation assistance. It is also important to specify any deviations from the basic system which are desired. In many cases, the addition of specific modules or options is of major consequence to the user. (2, p. 37)

The next issue of concern is the "buyer and seller" agreement to whatever constitutes delivery of the "software". For example, thirty days after the system is operational, when the software is first used, when training is completed, or what have you.

Most proprietary software is warranted; this usually means that the seller will fix bugs during the warranty period. The wise software buyer will clearly understand the warranty aspect of the particular vendor he is dealing with. Likewise, it is important that the buyer know what happens at the end of the warranty period. It is common practice for the vendor to offer an annual maintenance agreement. You know as well as I that the "completely de-bugged program" is a rarity and thus the maintenance agreement is a good insurance policy against a buyer having to maintain the program himself. The buyer of software should not be reluctant to enter into license agreements with vendors; he must, however, exercise good business judgment and be completely familiar with what he gets and the terms governing the transaction.

HOW GOOD IS IT?

It is well to understand the breadth of form a program acquired from the outside may take. This is quite a wide range and runs from "as-is" to "near-custom". The "as-is" mode is not generally applicable to application programs but is more applicable to utilities and operating system modifications. This is not to be interpreted to mean that no application packages are sold "as-is" -- in fact, many are. What is important is that the vendor is often reluctant to modify the basic logic of the package. You of course may run the risk of modifying it yourself and it may no longer be properly warranted.

Another level of program release occurs where users may modify the program by control card or other form of parameter entry. In this form, the program's execution is modified to more appropriately match the user's needs. This method of control of a program's features is, of course, limited in the sense that not all conceivable options can be provided for--which introduces the next level of program release--I/O customization. In this form, the basic program remains the same but input and output are modified to cause the software package to be more user-related in a particular environment.

The closest thing to a "scratch-built" system is the software package that consists of main-line logic and user-defined modules which serve to make the system unique to the user. This mode is often used for programs critical to a user's success in business. Sometimes, the modules may be pre-coded and they are selected by the user in the same manner as one would select part of an Erector Set to build a bridge.

Ultimately, the worth of the program will be judged by its ability to do the job it was acquired to do. In this regard, knowledge of the program and the job involved is critical. By knowing what it is that must be done and what capability you have acquired to do it, you are in a position to set about installing it for your users. Installing any system requires interaction with the users and the D/P department. Where the system installed becomes a part of the day-to-day, week-to-week operation of your enterprise, changes in procedures are never -- I repeat, never -- non-trivial.

A CHANGING WORLD

Up to now, we have addressed -- in general terms -- the problem of acquiring software. In closing, I'd like to emphasize that changes are taking place now and forces are at work that mitigate change. Let me give you an example:

You may recall, a few short years ago the notion of an open-shop operation was a mode of operation which permitted anyone to run his job on the computer by simply signing up for the time needed. I'm reminded of a computer room I once visited that has a baseball bat on the wall with a sign below it that read, "Computer Scheduling Device". The open shop gave way to the closed shop, where all work submitted and/or performed on the computer was under the control of skilled operators and a schedule set up often long in advance. Significantly, production work was just that. It was run without the programmer being in attendance. This was also true in the debugging runs. We are now on the doorstep of a major new method of operation -- the "Remote Shop" wherein a goodly portion of the work run on the computer is for a remote user. You may ask why this is important to evaluating and selecting software. Very simply, a good local batch job may not be a good remote job. Also, consider the advent of the multi-user job

in the context of, say, three school districts using the same application program tailored for its own needs but all three tailored copies run at the same center.

In closing, I would like to reiterate comments made earlier. "The Computer Center Management is being held more and more accountable and cost-effectiveness requires that you look closely at outside sources for software."

BIBLIOGRAPHY

1. Frombolz, H. J. Legal Protection of Computer Programs and Confidential Information. Proceedings of the COMPCON 73 IEEE, 1973, p. 105-108.

2. Welke, L. A. Contract Details called Vital and Impact Must Be Understood. Computer World, Feb. 28, 1973, p. 37.